

ARTÍCULOS DE TECNOLOGÍA &gt; FRONT END

# ¿Por qué usar 'em' en su CSS hoy?



sergio-lopes

19/05/2021

## ¿Cómo funcionan los ems?

La gran diferencia entre em y px es que em es una **medida relativa**. El valor se calcula siempre teniendo en cuenta el Font-size del padre. Esto significa que un elemento con Font-size: 2em; será el doble del tamaño de la fuente del padre, cualquiera que sea. Y así sucesivamente. El padre del padre del padre del padre... Es una gran multiplicación de valores hasta llegar a la raíz del documento, el. Si no tiene una fuente definida, la mayoría de los navegadores utiliza **16px como estándar** para el.

Esto significa que este código:

```
<html>
  <body>
    <main style="font-size: 1.5em">
      <h1 style="font-size: 2em">Titulo</h1>
      <p style="font-size: 0.75em">Texto.</p>
    </main>
  </body>
</html>
```

..acaba sendo equivalente a:

```
<html style="font-size: 16px">
  <body style="font-size: 16px">
    <main style="font-size: 24px">
      <h1 style="font-size: 48px">Titulo</h1>
      <p style="font-size: 18px">Texto.</p>
    </main>
  </body>
</html>
```

## Mito: ya no se puede acceder a em

La recomendación de que es mejor para la accesibilidad proviene de la época de los IEs antiguos, 6, 5.5. En estos navegadores, cuando el usuario tenía dificultades para ver, podía aumentar la fuente. En la práctica, la fuente base de era incrementada. En otras palabras, usaba 16px por estándar, pero con el zoom se convertía en 20px, por ejemplo.

Así, se recalculó el valor de todos los elementos que estaban con em y se aumentó el tamaño de la página. **Era una especie de zoom, fundamental para la accesibilidad.** Los elementos que estaban con px, sin embargo, no aumentaban de tamaño. Mala accesibilidad. Y entonces la tendencia del em.

Pero todos los navegadores modernos, incluido IE, ya no tratan el zoom de esa manera. **En la web moderna, el concepto de píxel también es relativo.** Escribir 1 px en CSS no significa necesariamente un píxel físico en la pantalla. Cuando el usuario hace zoom al 200%, por ejemplo, hoy en día el navegador aumenta todas las medidas, incluido el píxel. Eso significa que 1px en CSS se dibujará con el doble de píxeles físicos (2px x 2px en una pantalla común).

O sea, **no necesitamos em para la accesibilidad.**

## ¿Dónde tienen sentido el em entonces?

La gran ventaja del em, sin embargo, está en su aspecto de multiplicar los valores según los padres. Esto significa que puede cambiar el Font-size de un elemento y afecta a todos sus hijos. Es muy útil crear secciones en la página donde los elementos deben aumentar proporcionalmente entre sí: un componente, un widget.

Es decir, usamos em para facilitar la escrita de nuestro CSS. Por eso es bueno dominar el uso del em y sepa cómo usarlo en su código.

El em, por supuesto, no es para todos los escenarios. Vincula las medidas del hijo con el tamaño de letra del padre. Tocas al padre y el niño se ve afectado. A veces, esto no es deseado. Solo usamos em cuando hay una relación estructural entre el hijo y el padre y queremos que uno afecte al otro. Donde no tenga sentido, sigue usando px.

## El famoso caso de los diseños responsivos.

El em dio un giro triunfal en diseños responsivos. Es muy común que quieras una fuente más pequeña en el móvil, que tenga menos espacio visible, pero una fuente más grande en el Desktop, más espacioso. Y desea que toda la página crezca proporcionalmente. Entonces el em.

Hacemos algo como esto:

```
@media (min-width: 500px) {  
  html {  
    font-size: 1.25em;  
  }  
}
```

```
@media (min-width: 800px) {  
  html {  
    font-size: 1.5em;  
  }  
}
```

```
@media (min-width: 1000px) {  
  html {  
    font-size: 1.75em;  
  }  
}
```

```
@media (min-width: 1200px) {  
  html {
```

```
font-size: 2em;  
}  
}
```

A medida que aumenta el tamaño de la pantalla, aumentamos el valor de em base en la tag. Esto tiene un efecto dominó en todos los elementos secundarios si están escritos con em.

Es muy bueno usar em en sitios responsivos para ajustar, a través de CSS, el tamaño de todos los elementos proporcionalmente.

Pero em no es una medida flexible ni más accesible. Al menos no en el sentido de otros como el porcentaje, que se adapta automáticamente. El em es fijo, solo el valor se calcula desde una cuenta más complicada.

## Bono: el rem y otras medidas verdaderamente flexibles

El rem es una nueva medida similar. La diferencia es que la cuenta no toma en consideración a todos los padres, sino solo la raíz, el. Eso significa que jugar con el Font-size de cualquier elemento de la página no reflejará ni obstaculizará a otros. Realmente solo si movemos el. Es útil en el caso del diseño responsivo que vimos antes.

Pero recuerda que multiplicar el em por el Font-size del padre es algo bueno en muchas situaciones. Varias partes de la página pueden verse como pequeños componentes autónomos. Y cambiar el padre del componente tiene que afectar a todos los hijos. El em es excelente para esto, a diferencia del rem.

Y, si deseas una medida verdaderamente flexible y adaptable para las fuentes, la respuesta está en las nuevas **viewport units**. En particular, un código como `html { font-size: 2vw; }` hará que la fuente se adapte automáticamente a cualquier tipo de resolución. Pero eso es tema para otro artículo.

Si deseas saber más sobre medidas flexibles, buenas prácticas de CSS y también de diseño responsivo, no te pierdas los [cursos de front-end de Alura](#).

¿Y a ti te gusta usar em?

Puedes leer también:

- [Tu código CSS puede ser más limpio, flexible y reutilizable.](#)

- [Empezando a organizar tu CSS](#)
- [¿Cómo lidiar con los límites de resolución en sitios responsivos?](#)

ARTÍCULOS DE TECNOLOGÍA > FRONT END

## En Alura encontrarás variados cursos sobre Front End. ¡Comienza ahora!

**SEMESTRAL**

**US\$49,90**

un solo pago de US\$49,90

- ✓ 218 cursos
- ✓ Videos y actividades 100% en Español
- ✓ Certificado de participación
- ✓ Estudia las 24 horas, los 7 días de la semana
- ✓ Foro y comunidad exclusiva para resolver tus dudas
- ✓ Acceso a todo el contenido de la plataforma por 6 meses

**¡QUIERO EMPEZAR A ESTUDIAR!**

[Paga en moneda local en los siguientes países](#)

**ANUAL**

**US\$79,90**

un solo pago de US\$79,90

- ✓ 218 cursos
- ✓ Videos y actividades 100% en Español
- ✓ Certificado de participación
- ✓ Estudia las 24 horas, los 7 días de la semana
- ✓ Foro y comunidad exclusiva para resolver tus dudas
- ✓ Acceso a todo el contenido de la plataforma por 12 meses

**¡QUIERO EMPEZAR A ESTUDIAR!**

[Paga en moneda local en los siguientes países](#)

Acceso a todos  
los cursos

Estudia las 24 horas,  
dónde y cuándo quieras

Nuevos cursos  
cada semana

## NAVEGACIÓN

PLANES

INSTRUCTORES

BLOG

POLÍTICA DE PRIVACIDAD

TÉRMINOS DE USO

SOBRE NOSOTROS  
PREGUNTAS FRECUENTES

## ¡CONTÁCTANOS!

¡QUIERO ENTRAR EN CONTACTO!

## BLOG

PROGRAMACIÓN  
FRONT END  
DATA SCIENCE  
INNOVACIÓN Y GESTIÓN  
DEVOPS

AOVS Sistemas de Informática S.A  
CNPJ 05.555.382/0001-33

## SÍGUENOS EN NUESTRAS REDES SOCIALES



## ALIADOS



En Alura somos unas de las Scale-Ups seleccionadas por Endeavor, programa de aceleración de las empresas que más crecen en el país.



Fuimos unas de las 7 startups seleccionadas por Google For Startups en participar del programa Growth Academy en 2021

POWERED BY



## CURSOS

### Cursos de Programación

Lógica de Programación | Java

### Cursos de Front End

HTML y CSS | JavaScript | React

### Cursos de Data Science

Data Science | Machine Learning | Excel | Base de Datos | Data Visualization | Estadística

### Cursos de DevOps

Docker | Linux

### Cursos de Innovación y Gestión

Productividad y Calidad de Vida | Transformación Ágil | Marketing Analytics |

Liderazgo y Gestión de Equipos | Startups y Emprendimiento